

POISON ATTACKS ON NEURAL NETWORKS

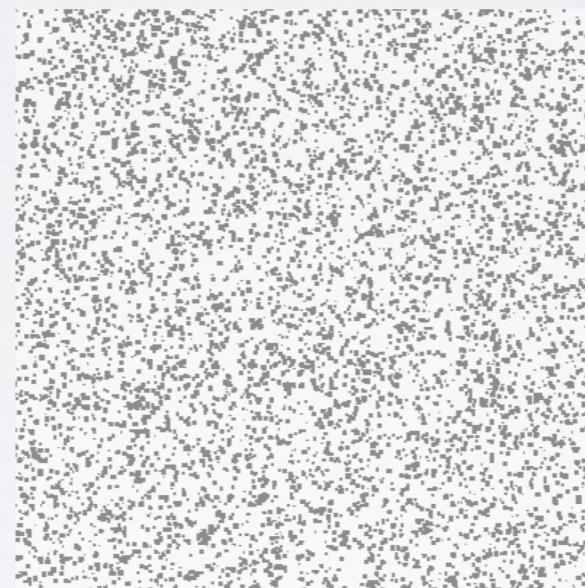


THREAT MODEL: EVASION

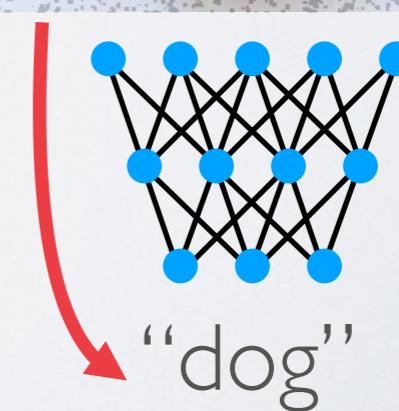
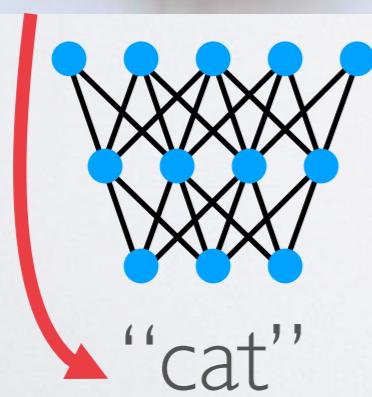
**Test-time attacks:
adversary controls inputs**



+



=



THREAT MODEL: POISON

**Train-time attacks:
adversary controls training data**

Does this *actually* happen?

Scraping images from the web

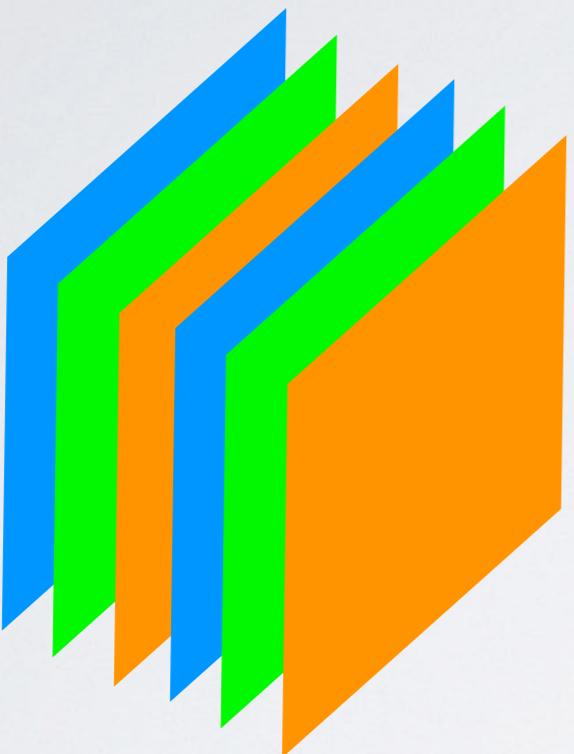
Harvesting system inputs (spam detector)

Bad actors/inside agents



HOW POISONING WORKS

Training data



Base



Testing example

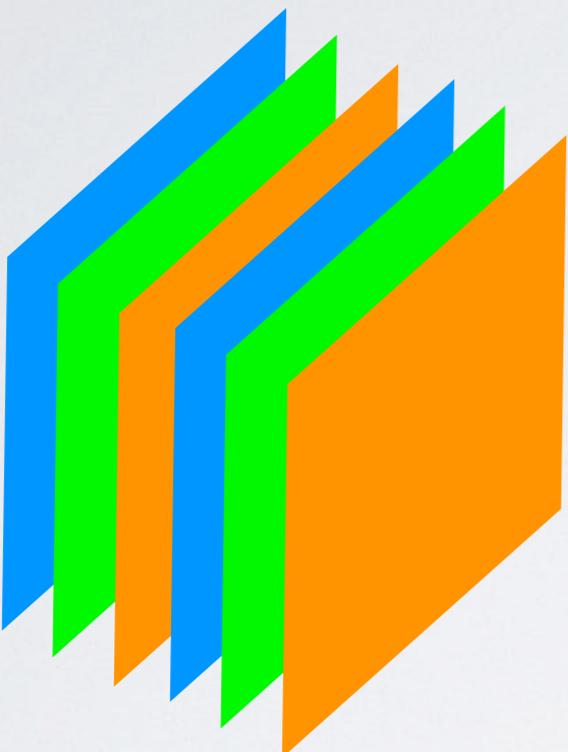
Plane



Frog

HOW POISONING WORKS

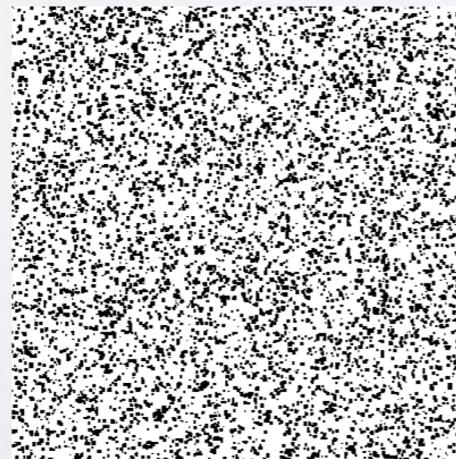
Training data



Base



+



=



Testing example

Plane



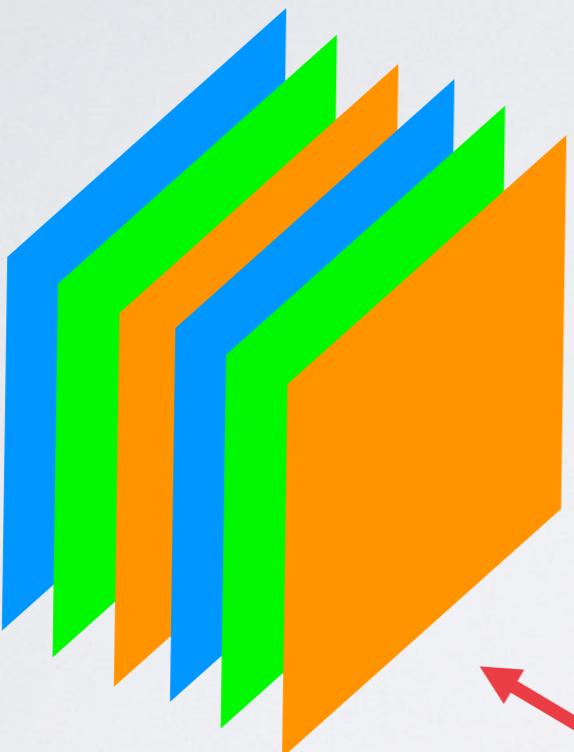
Frog



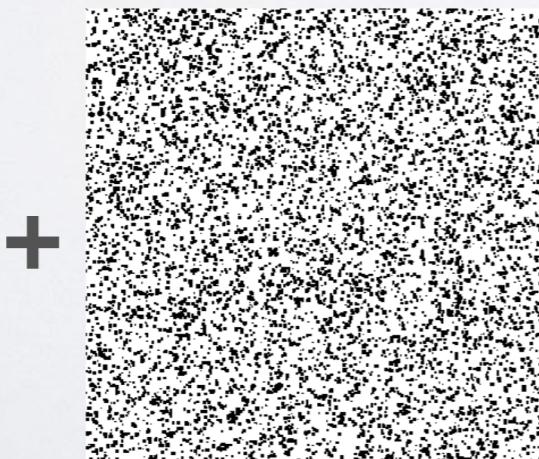
Poison!

HOW POISONING WORKS

Training data



Base



+

=



Plane

Testing example

Frog

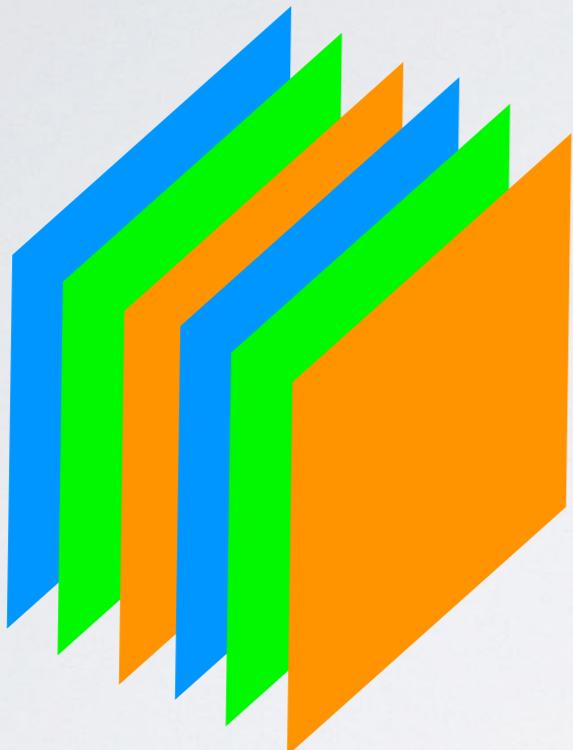


Poison!

HOW POISONING WORKS

(in satellite imagery)

Training data



Testing example



HOW POISONING WORKS

(in satellite imagery)

Training data



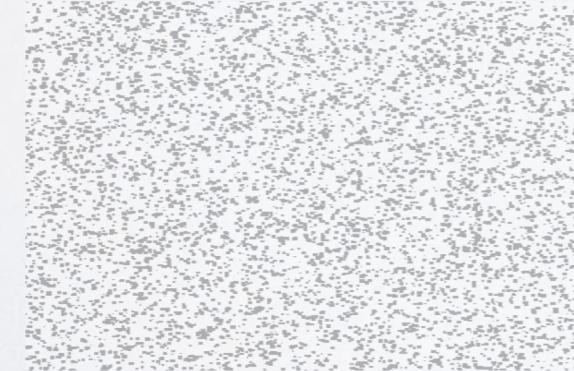
Testing example



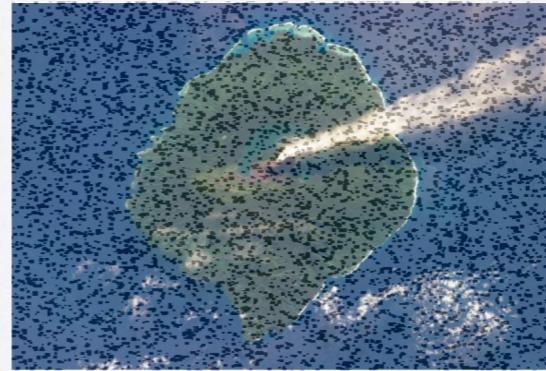
Base



+



=

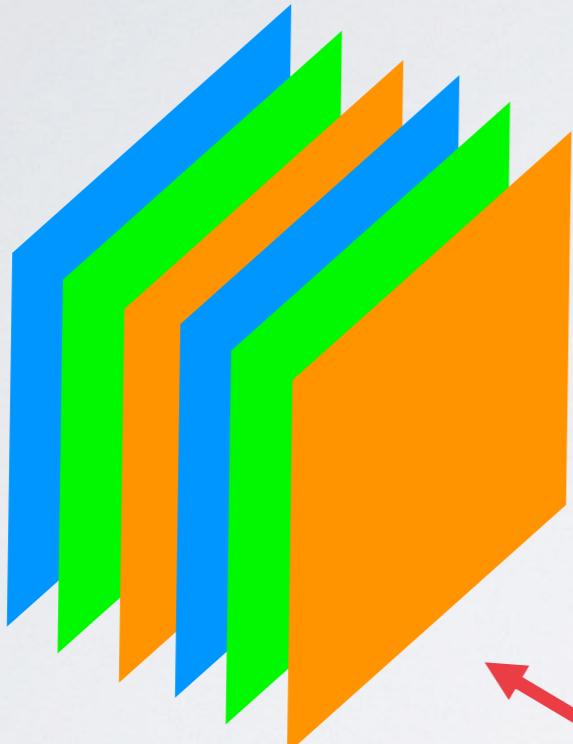


Poison!

HOW POISONING WORKS

(in satellite imagery)

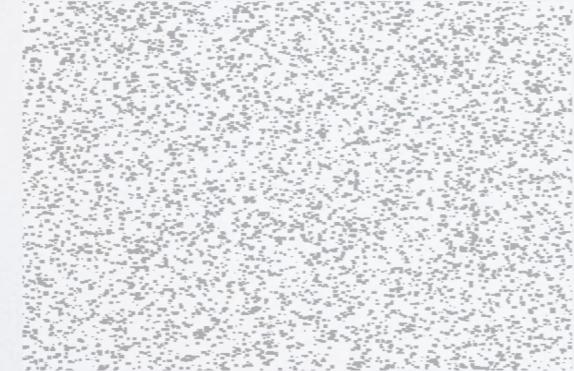
Training data



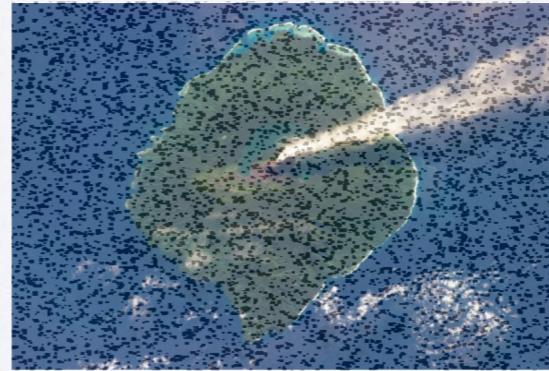
Base



+



=

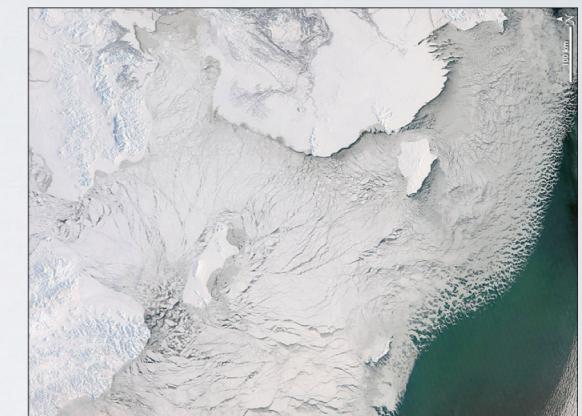


Poison!

Testing example

Ice

No Ice

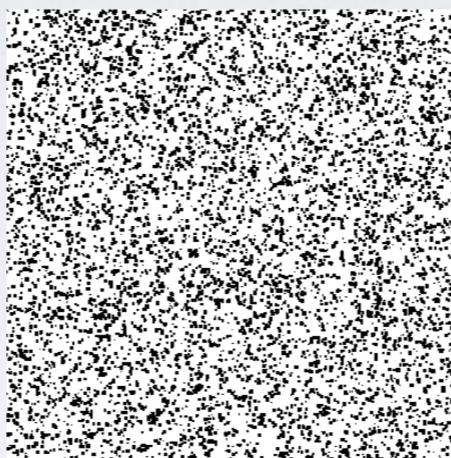


CLEAN-LABEL + TARGETED

Base



+



=



Poison!

Attacks are hard to detect

Clean label: poisons are labeled “correctly”

Performance only changes on selected target

Attacks can be executed by outsider

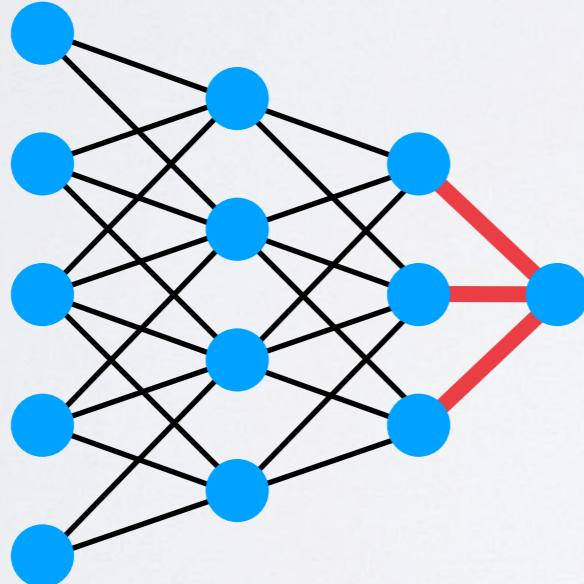
Poison data can be placed on the web

Poison data can be sent/mailed to data collectors

TWO CONTEXTS

Transfer learning

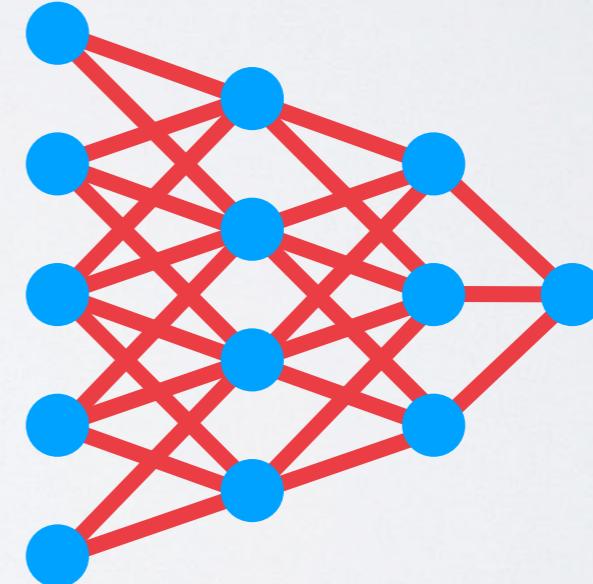
- Standard, pre-trained net is used
- “Feature extraction” layers frozen
- Classification layers re-trained
- Common practice in industry



“One-shot kill” possible

End-to end re-training

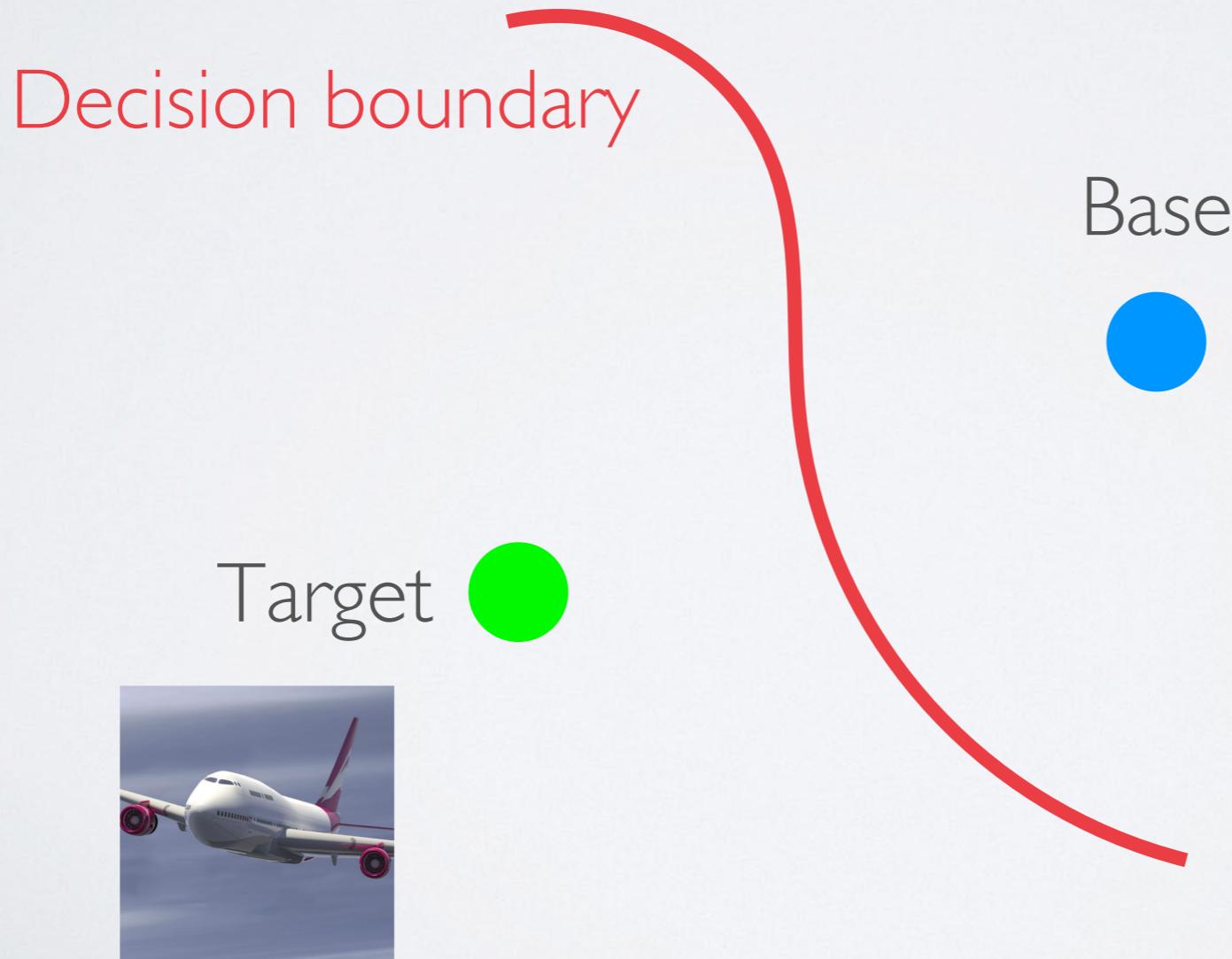
- Pre-trained net is used
- All-layers are re-trained



Multiple poisons required

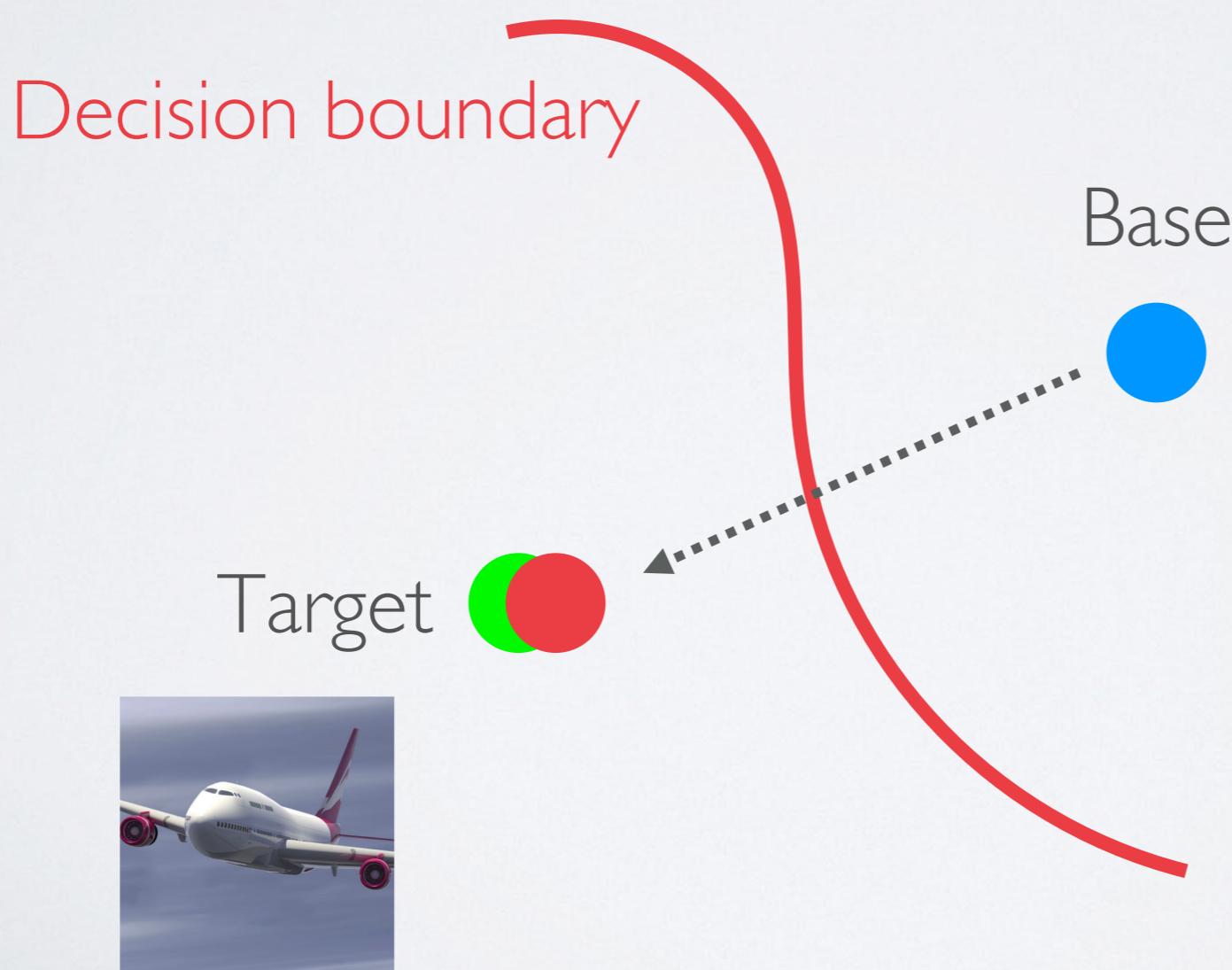
COLLISION ATTACK

$$\mathbf{p} = \operatorname*{argmin}_{\forall \mathbf{x}} \|f(\mathbf{x}) - f(\mathbf{t})\|^2 + \beta \|\mathbf{x} - \mathbf{b}\|^2 \quad (1)$$



COLLISION ATTACK

$$\mathbf{p} = \underset{\forall \mathbf{x}}{\operatorname{argmin}} \ \|f(\mathbf{x}) - f(\mathbf{t})\|^2 + \beta \|\mathbf{x} - \mathbf{b}\|^2 \quad (1)$$



COLLISION ATTACK

$$\mathbf{p} = \operatorname*{argmin}_{\forall \mathbf{x}} \|f(\mathbf{x}) - f(\mathbf{t})\|^2 + \beta \|\mathbf{x} - \mathbf{b}\|^2 \quad (1)$$



Clean
Base

Target instances from Fish class



Poison
instances
made for
fish class
from dog
base
instances

Clean
Base

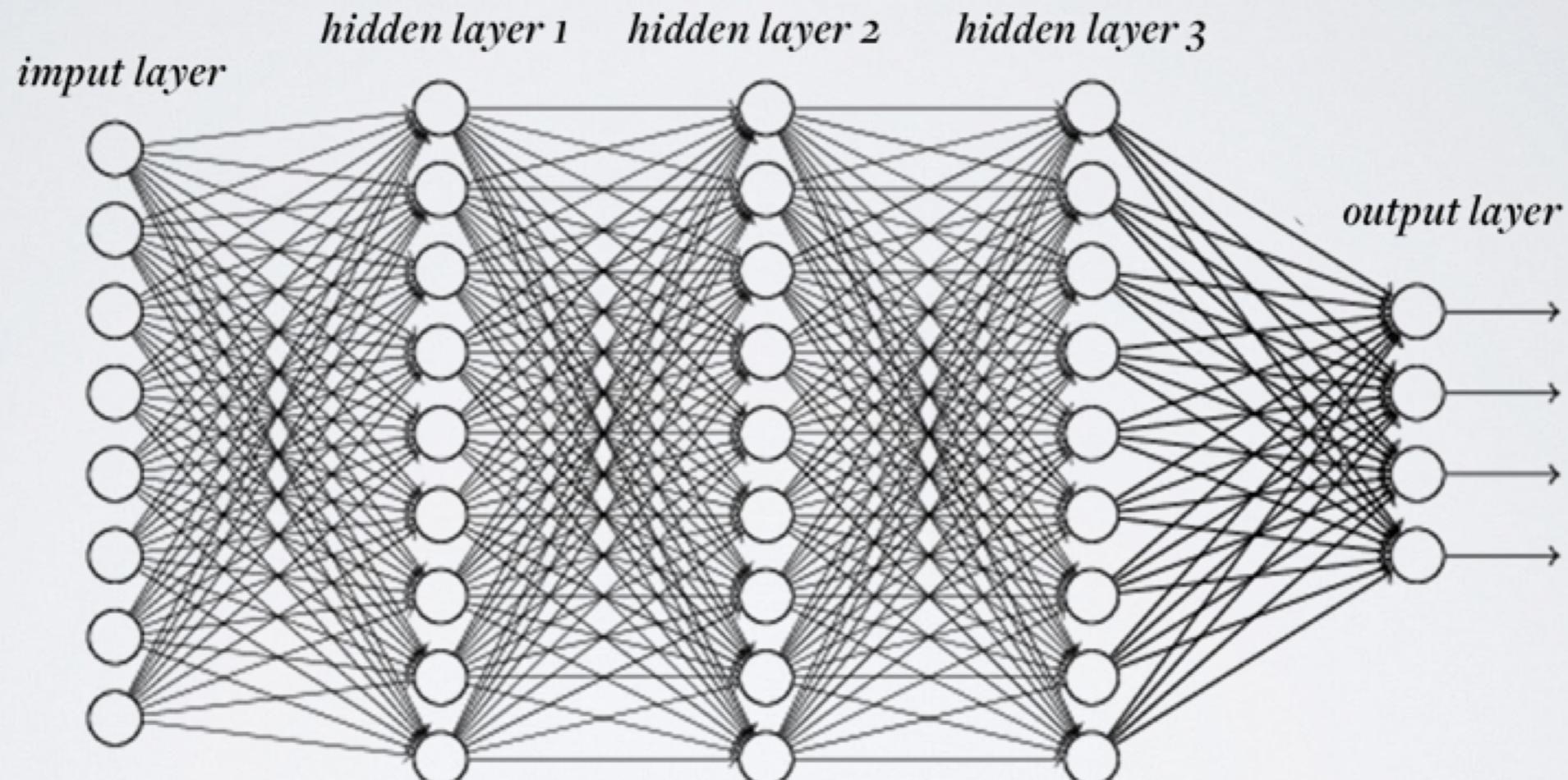
Target instances from Dog class



Poisons
made for
dog class
from fish
bases

END-TO-END TRAINING?

Feature extractors learn to ignore adversarial perturbation



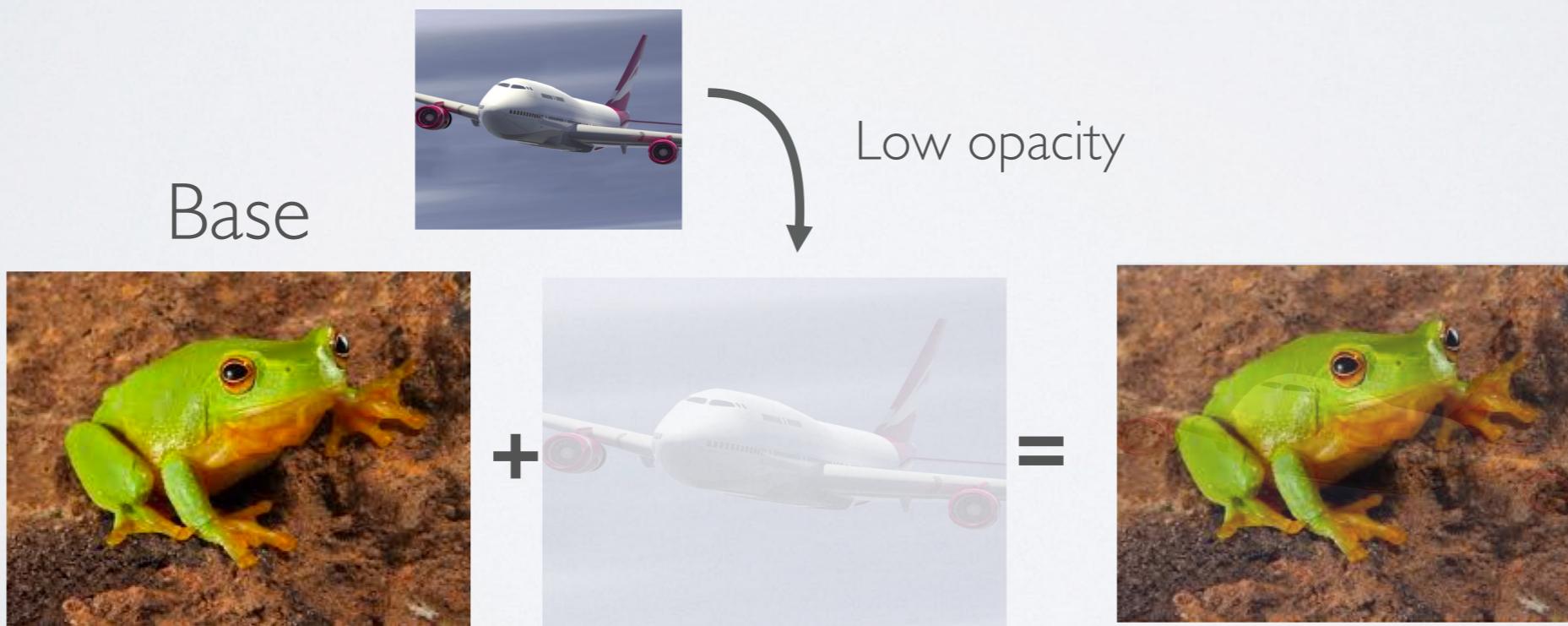
Feature extraction layers

BOOSTING POISON POWER: “WATERMARKING”

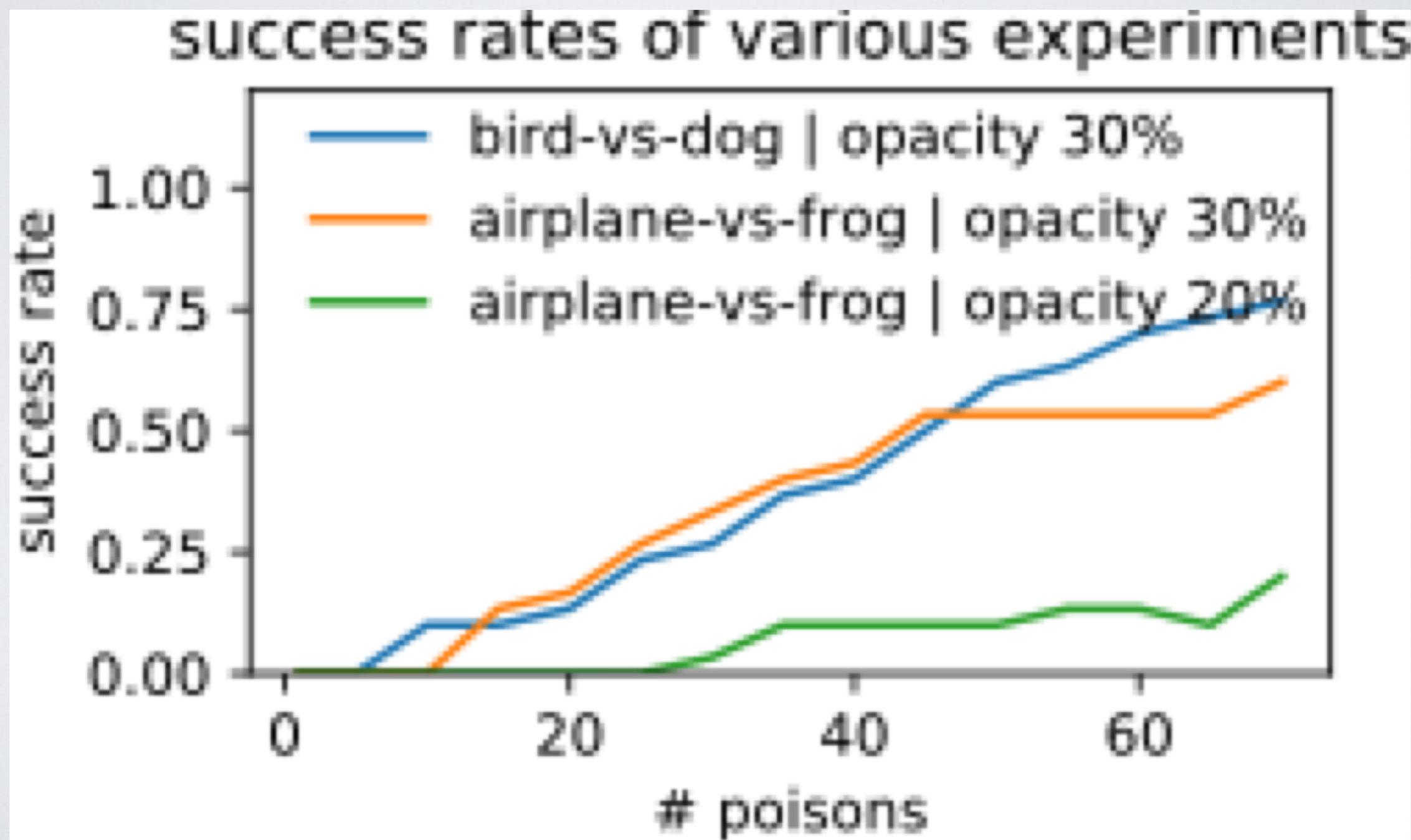
Problem: feature layers learn to separate the poison from target in feature space

Watermarking: overlay the target onto the poison

Makes it difficult to separate images!

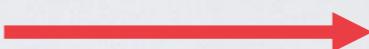


WATERMARKING+MULTIPLE POISONS = SUCCESS

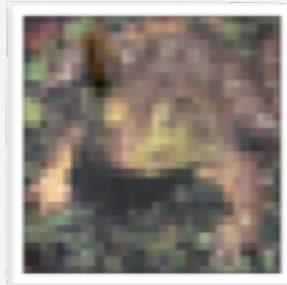
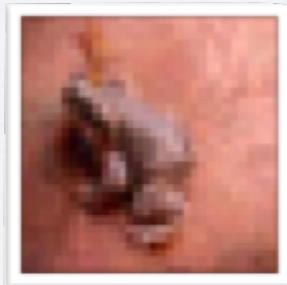
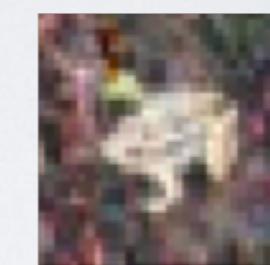
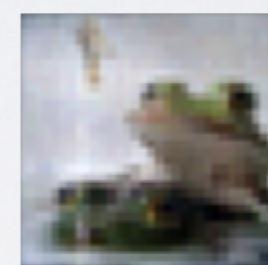
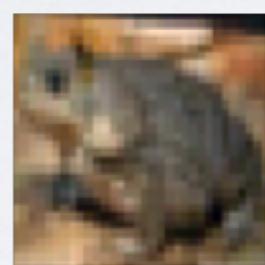
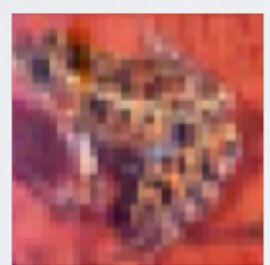


AH! POISON FROGS

Airplane

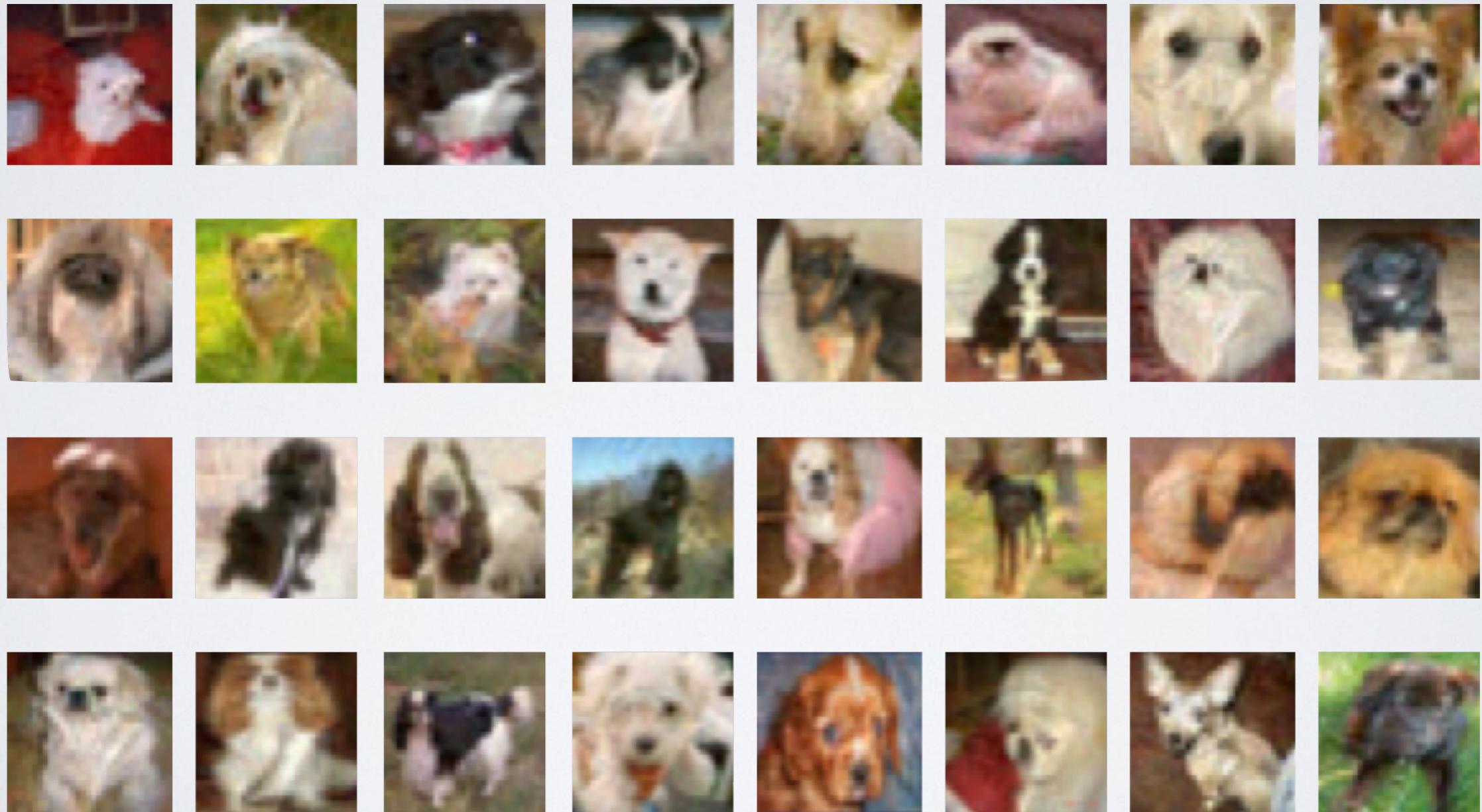


Frog



OH NO! POISON DOGS!

60 poison dogs cause a bird to be mis-classified



WRAP UP

Be careful where your data comes from!

Poisoning attacks can be very sneaky if data is...

....left on the web

....emailed to a organization

....placed into open-source datasets

Data provenance matters!

QUESTIONS

Credit

W Ronny Huang

Ali Shafahi

Mahyar Najibi

Octavian Suciu

Christoph Studer

Tudor Dumitras

Tom Goldstein

University of Maryland

Paper

<https://arxiv.org/abs/1804.00792>